

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-016355

(43)Date of publication of application : 17.01.1997

(51)Int.Cl. G06F 3/12
B41J 2/485
B41J 5/44

(21)Application number : 07-160722 (71)Applicant : CANON INC

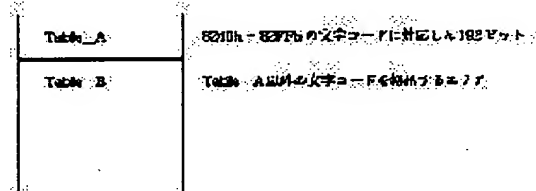
(22)Date of filing : 27.06.1995 (72)Inventor : NAKAGIRI KOJI

(54) INFORMATION PROCESSOR AND STORAGE MEDIUM FOR STORING FONT DOWN LOAD MANAGING PROGRAM

(57)Abstract:

PURPOSE: To enable the down load of character having the high frequency of usage at all times, to shorten the time for judging whether the down load is completed or not and to reduce the memory size of a managing table by providing a judging means for judging the usage frequency of font and a selecting means for selecting either a managing table Table-A or a Table-B corresponding to whether the usage frequency is high or low.

CONSTITUTION: A character registration managing table to be managed by a printer driver is used. Inside a RAM, the character registration managing table is provided with a registered character code managing Table-A for font face name, character size, character modification data and high usage frequency characters having several pieces of data concerning a character set and a registered character code managing Table-B for low usage frequency characters. When it is judged that the font to be printed has the high frequency of usage, the Table-A is selected but when it is judged that the font to be printed has the low frequency of usage, the Table-B is selected.



LEGAL STATUS

[Date of request for examination] 27.06.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3618828

[Date of registration] 19.11.2004

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's

BEST AVAILABLE COPY

décision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The information processor carry out having a selection means will choose the 1st managed table if it judges that it is the information processor which downloads a font to an airline printer, and the font which should print with a decision means judge the operating frequency of the font which should be printed with said airline printer, and said decision means is a font with high operating frequency, and will choose the 2nd managed table if it judges that said font which should print is a font with low operating frequency with said decision means as the description.

[Claim 2] Said decision means is an information processor according to claim 1 characterized by making into the decision criterion as a font coding region with high operating frequency the font coding region which the single continuous font coding region or plurality followed.

[Claim 3] Said 1st managed table is an information processor according to claim 1 with which a font with the high operating frequency which should be printed with said airline printer is characterized by managing the information which shows whether it is download ending by 1 bit to said airline printer.

[Claim 4] Said 2nd managed table is an information processor according to claim 1 with which a font with the low operating frequency which should be printed with said airline printer is characterized by managing the information which shows whether it is download settled by the data of character code length to said airline printer.

[Claim 5] The information processor according to claim 1 characterized by judging whether said font which should be printed is download settled at said airline printer using the 1st managed table or the 2nd managed table chosen with said selection means.

[Claim 6] The decision process which is the storage which memorizes the font download manager performed with the information processor which downloads a font to an airline printer, and judges the operating frequency of the font which should be printed with said airline printer, If it judges that said font which should be printed is a font with high operating frequency at said decision process, the 1st managed table will be chosen. The storage characterized by memorizing a font download manager equipped with the selection process which will choose the 2nd managed table if it judges that said font which should be printed is a font with low operating frequency at said decision process.

[Claim 7] Said decision process is a storage according to claim 6 characterized by making into the decision criterion as a font coding region with high operating frequency the font coding region which the single continuous font coding region or plurality followed.

[Claim 8] Said 1st managed table is a storage according to claim 6 with which a font with the high operating frequency which should be printed with said airline printer is characterized by managing the information which shows whether it is download settled by 1 bit to said airline printer.

[Claim 9] Said 2nd managed table is a storage according to claim 6 with which a font with the low operating frequency which should be printed with said airline printer is characterized by managing the information which shows whether it is download settled by the data of character code length to said airline printer.

" [Claim 10] The storage according to claim 6 characterized by judging whether said font which should be printed is download settled at said airline printer using the 1st managed table or the 2nd managed table chosen at said selection process.

[Claim 11] A font download manager equipped with said decision process and said selection process is a storage according to claim 6 characterized by being external storage removable to said information processor.

[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] This invention is a storage which memorizes the font download manager performed with an information processor and the information processors concerned, such as a host computer which downloads a font to an airline printer.

[0002]

[Description of the Prior Art] The environment of WYSIWYG (What You See Is What You Get) which prints to a printer by the same design as the alphabetic character displayed on a display is realized by having the device which generates an alphabetic character image data in a host computer, and creating and transmitting the alphabetic character image data doubled with each resolution to a display and a printer in recent years. It is necessary to transmit the image data of an alphabetic character to a printer in such a printing environment. In case it prints in this environment conventionally, the method (character image drawing) of transmitting an alphabetic character image data for every single character or the alphabetic character image data transmitted once was memorized in the printer, and the method (character image download) of reusing the image data memorized when the same alphabetic character was printed has been used.

[0003] The management method which identifies the alphabetic character which uses as one character set the alphabetic character group which makes the same the attributes (a font name, a character size, cell size, character decoration, etc.) of an alphabetic character by the approach of printing by this character image download, and is contained in that character set by the character code is common. In this case, one alphabetic character image data will be expressed in the combination of a character set and a character code.

[0004] In order to manage alphabetic character printing by said download from a host computer side, the data of being finishing [the alphabetic character image data of an alphabetic character which is going to print / download] already are needed. For this reason, the data in which it is shown for every character set whether it is finishing [download of each alphabetic character] are managed. The data area which stores the data in which it is shown hereafter whether it is finishing [this download] will be called a registered character code managed table. At least 1 bit which usually shows [download ending or] whether that is right to each character code is required for this data. Or although there is also a method of downloading beforehand all the alphabetic characters contained in the character set, since the amount of data becomes huge in an alphabetic coded character set as which a character code is expressed by two or more bytes, it is not realistic. Therefore, said management is needed in order to perform Japanese alphabetic character download.

[0005] Although the management memory [finishing / download] which is needed on a host in order to realize this is at least 1 bit per each alphabetic character, when it prepares the memory (from thousands to 10,000 [The case of Japanese] thousands of bits) of the number of bits corresponding to all character codes for every character set, it will press the memory by the side of a host. Since there is generally a limitation in the RAM capacity for memorizing on the printer in fact when printing by download, the number of alphabetic characters which can usually

be registered is restricted. For this reason, the number of the maximum registered alphabetic characters was conventionally prepared for every character set, the memory which stores the character code for several of those alphabetic character minutes was prepared, and downloaded character code management has been performed by the approach of keeping as a list the character code downloaded there. and a character image downloads — ** — finishing [it is alike and / download] — it is — the data which recorded the data in which things are shown on a registered character code managed table, stopped printing by download, changed and outputted to other printing approaches (for example, image drawing) when the condition that it is not downloadable during printing is reached, or have already downloaded deleted, and printing has been conventionally performed by the approach of downloading to the vacant part.

[0006]

[Problem(s) to be Solved by the Invention] However, by this conventional approach, after reaching the upper limit of the number of alphabetic characters which can be registered, the problem that the image data of an alphabetic character with high operating frequency was not newly downloadable might occur.

[0007] Moreover, in order to manage a character code [finishing / download] by the list, even if it was the case of an alphabetic character with high operating frequency, the problem of taking time amount was in retrieval on the occasion of the judgment of being finishing [download], and record.

[0008] Moreover, when an alphabetic character registration managed table was created so that the character code downloadable for every character set for several maximum alphabetic character minutes can be stored, the memory size on the host who uses it became large, and there was a problem that it cannot secure.

[0009] This invention was made in order to solve these troubles, and it aims at always enabling download of an alphabetic character with high operating frequency.

[0010] Moreover, it aims at shortening the time amount required in order to be related with an alphabetic character with high operating frequency and to judge or record whether it is finishing [download].

[0011] Furthermore, it aims at lessening memory size which a registered character code managed table needs.

[0012]

[Means for Solving the Problem] It has a decision means the information processor concerning this invention is the information processor which downloads a font to an airline printer, and judge the operating frequency of the font which should print with said airline printer, and a selection means will choose the 2nd managed table if the 1st managed table will choose if it judges that said font which should be printed is a font with high operating frequency with said decision means, and said font which should print judges that it is a font with low operating frequency with said decision means.

[0013] Moreover, the storage concerning this invention is a storage which memorizes the font download manager performed with the information processor which downloads a font to an airline printer. The decision process which judges the operating frequency of the font which should be printed with said airline printer, If it judges that said font which should be printed is a font with high operating frequency at said decision process, the 1st managed table will be chosen. If it judges that said font which should be printed is a font with low operating frequency at said decision process, a font download manager equipped with the selection process which chooses the 2nd managed table will be memorized.

[0014]

[Function] If it judges that the font which should choose the 1st managed table and should be printed is a font with low operating frequency if it judges that the font which should be printed in this invention by making it the above-mentioned configuration is a font with high operating frequency, the 2nd managed table will be chosen.

[0015]

[Example] Next, the suitable example of this invention is explained to a detail with reference to a drawing.

[0016] Drawing 1 is the block diagram of the print control unit which is one example of this invention. In addition, if the function of this invention is performed, even if it is the system by which processing is performed through networks, such as LAN, even if it is the system which consists of two or more devices even if it is the device of a simple substance, it cannot be overemphasized that this invention is applicable.

[0017] In drawing, 3000 is a host computer, it has CPU1 which performs the document processing system in which the graphic form, the image, the alphabetic character, the table (a spreadsheet etc. is included), etc. were intermingled based on the document processing system program memorized by ROM for a program or external memory 11 of ROM3, and CPU1 controls each device connected to a system bus 4 in the gross.

[0018] Moreover, to ROM for a program or external memory 11 of this ROM3, the operating system program (henceforth, OS) which is a control program of CPU1 is memorized, the font data used for ROM for fonts or external memory 11 of ROM3 in the case of the above-mentioned document processing system is memorized to it, and ROM for data of ROM3 memorizes to it the various data used in case the above-mentioned document processing system etc. is performed. 2 is RAM and functions as the main memory of CPU1, a work area, etc. 5 is a keyboard controller (KBC) and controls the key input from a keyboard 9 or a non-illustrated pointing device. 6 is a CRT controller (CRTC) and controls the display of CRT display (CRT) 10. 7 is a disk controller (DKC) and controls access with the external memory 11 which memorizes a boot program, various kinds of applications, font data, a user file, an edit file, an alphabetic character image-data generator (following font writer riser), a printer control command generator (following printer driver), etc., such as a hard disk (HD) and a floppy disk (FD). 8 is a printer controller (PRTC), and it connects with a printer 1500 through the predetermined bidirection interface (interface) 21, and it performs communications control processing with a printer 1500. In addition, CPU1 performs expansion (writer rise) processing of the outline font to the display information RAM set up for example, on RAM2, and makes WYSIWYG on CRT10 possible. Moreover, CPU1 opens the various windows registered based on the command directed by the mouse cursor which is not illustrated on CRT10, and performs various data processing.

[0019] In a printer 1500, 12 is Printer CPU and outputs the picture signal as a print-out to the printing section (printer engine) 17 connected to a system bus 15 based on the control program memorized by external memory 14, such as a control program memorized by ROM for a program of ROM13. Moreover, the control program of CPU12 etc. is memorized for the program ROM of this ROM13. The font data used in case the above-mentioned print-out is generated to ROM for fonts of ROM13 was memorized, and in being the printer no external memory 14, such as a hard disk, is [printer] in ROM for data of ROM13, it has memorized the information used on a host computer. The communications processing with a host computer of CPU12 has become possible through the input section 18, and the host computer 3000 constitutes the information in a printer etc. possible [a notice]. 19 is RAM which functions as the main memory of CPU12, a work area, etc., and it is constituted so that memory space can be extended with the option RAM connected to the extension port which is not illustrated. In addition, RAM19 is used for a print-out expansion field, an environmental data storage field, NVRAM, etc. The external memory 14 mentioned above, such as a hard disk (HD) and an IC card, has access controlled by the memory controller (MC) 20. It connects as an option and external memory 14 memorizes font data, an emulation program, form data, etc. Moreover, a switch, an LED drop, etc. for actuation are allotted with the control panel which 1501 mentioned above.

[0020] Moreover, the external memory mentioned above may be constituted so that not only one piece but two or more external memory which stored the program which interprets the printer control language with which it has at least one or more pieces, and an option font card differs from a language system in addition to an internal-organs font can be connected. Furthermore, it has NVRAM which is not illustrated and you may make it memorize the printer mode setting information from a control panel 1501.

[0021] using the flow chart which shows hereafter the control program performed by CPU1, it is alike in detail and one example of this invention is described.

[0022] Drawing 2 is a flow chart which shows the general procedure in the printer driver which downloads an alphabetic character image data. First, the outline of the procedure shown by drawing 2 is described. At step 2.1, printing data are received from the operating system (OS) or application on the host computer which has controlled the printer driver. At step 2.2, it judges whether it is already registered to a printer in the image data of the alphabetic character shown by the printing data acquired at step 2.1. When judged with it being registered, it shifts to the processing (step 2.7) which generates the instruction which prints the registered alphabetic character. When judged with un-registering, an alphabetic character image data is acquired from Font ROM based on the printing data obtained at step 2.1 (step 2.3). At step 2.4, the printer command which registers into a printer the character image acquired at step 2.3 is generated, and it transmits to (step 2.4) and a printer (step 2.5). At step 2.6, the flag which shows that it is already registered to a printer in the alphabetic character shown by printing data is set. At step 2.7, the command which prints the alphabetic character already registered to the printer is generated, and it transmits to a printer (step 2.8).

[0023] It is the parts of the judgment processing in step 2.2, and the set of the registered flag in step 2.6 that the alphabetic character image-data management method which carries out this invention is enforced in such procedure of a series of.

[0024] Next, the part in which this invention is carried out is explained in detail. The flow chart which shows the procedure which judges whether it is registered in step 2.2 of drawing 2 is shown in drawing 3. In this processing, the alphabetic character registration managed table managed by the printer driver which carries out this invention is used. Here, although a font environment which an alphabetic character decides by Code, FaceName, Style, and Point is assumed, sufficient data to specify an alphabetic character according to an environment should just be contained. And FaceName, Style, and all Point(s) manage a match as one character set, and the example which uses one alphabetic character registration managed table to the alphabetic character belonging to the same character set is shown here. One alphabetic character registration managed table is constituted from following contents which have some data about a character set besides a registered character code managed table by RAM2.

[0025] Alphabetic-character registration managed table: FaceName Point Style Table_A Table_B -
- (data 1)

FaceName: Font face name Point: character-size Style: Initialize the variable used by this processing at the registered character code managed table step 3.1 for alphabetic characters with the low registered character code managed table Table_B: use frequency for alphabetic characters with high character decoration data Table_A: use frequency. At step 3.2, it investigates whether the last alphabetic character registration managed table was reached. At step 3.3, it judges [whether it is in agreement with FaceName of the printing data which all of FaceName of the alphabetic character registration managed table Table [i], Style, and Point acquired at step 2.1, Style, and Point, and] whether in investigating and being in agreement, the character code shows the alphabetic character with high operating frequency (step 3.4). However, Code is a shifted JIS code and assumes 8240h-82FFh to be the character code field where operating frequency is high here. When judged with operating frequency being high here, Table_A [Code 8240h] is investigated (step 3.6), in ON, it judges with it being registered, and processing is ended. In OFF, it judges with un-registering, and processing is ended in it. In investigating whether Table_B [x] which makes Code a value exists in Table_B when judged with operating frequency being low at step 3.4 (step 3.7) and existing, it judges with it being registered and ends processing. In not existing, it judges with un-registering and ends processing. When the data of Table [i] differ from each data of printing data at step 3.3, it shifts to the processing about the following alphabetic character registration managed table (step 3.5). And when the match of each data does not exist even to the last alphabetic character registration managed table, it judges with un-registering (step 3.8).

[0026] Here, it explains in more detail about Table_A and Table_B. It is shown whether these tables already have the registered image data of each alphabetic character to a printer. The contents are shown in drawing 4. The table adds the downloaded character code to the list, and serves as partial Table_A which expresses registered and un-registering by 1 bit to each

character code from registered and partial Table_B which carries out a non-registered expression. Table_A is a part currently assigned 1 bit at a time to the alphabetic character with high operating frequency. Here, 192 characters which the character code of 81FFh(s) followed from 8140h containing a full-size alphanumeric hiragana in the example of a shifted JIS code are assigned to Table_A. Generally, if it is a Japanese document, the operating frequency of a hiragana is high and alphabetic character kinds, such as a hiragana, are assigned to the character code field which continued by many character coding schemes. Therefore, if a character coding scheme exists in the character code field to which the alphabetic character with high operating frequency continued, it can apply not only Shift JIS but this invention. Moreover, although the hiragana was mentioned as the example, it cannot be overemphasized that it can carry out also to alphabetic character kinds, such as katakana and a notation (punctuation etc. is included). Moreover, it is also possible to assign a single byte code to the field of Table_A and to assign 2 bytes (or more than it) of code to the field of Table_B. Table_B is a field for recording having downloaded the alphabetic character image data of the character code which does not belong to Table_A, for example, if it is used for a Japanese 2-byte code, it will initialize the field of Table_B with the suitable value (for example, 0) to the coding scheme, and it creates a downloaded character code list by adding the character code which downloaded to the part as for which Table_B was vacant. In addition, the size of this field doubles the number of alphabetic characters managed by Table_A and Table_B, and it is made to become a part for an alphabetic character downloadable for every character set.

[0027] Thus, compared with the conventional method of storing all download character codes on memory, when the alphabetic character image data investigates the data (flag) in which it is shown to a printer whether it is finishing [download] about an alphabetic character with high operating frequency by creating an alphabetic character registration managed table, since a flag can be uniquely found from a character code, the time amount which a judgment takes is short and ends. Moreover, since the flag of an alphabetic character with high operating frequency is 1 bit each, it can lessen memory usage of the whole used for download management compared with the case where the memory of several downloadable alphabetic character minute character code storing is used. In addition, about Table_B, it is necessary to search for a character code as usual. A character code is judged a numeric value and this part is searched at a high speed with the application of the Hash Function which is well-known technique. In addition, if a Hash Function is used at the time of retrieval, also in case it records download ending, it is necessary to store a character code in Table_B using a Hash Function.

[0028] At step 2.6 of drawing 2, the flag of the alphabetic character registered into the alphabetic character registration managed table to the printer at step 2.5 is set. Drawing 5 shows the flow chart of this processing. A variable required of this processing is initialized at step 5.1. At step 5.2, it investigates whether the last alphabetic character registration managed table was reached. At step 5.3, it investigates whether in when [which are in agreement with FaceName of the printing data which all of FaceName of the alphabetic character registration managed table Table [i], Style, and Point acquired at step 2.1, Style, and Point / or or when in agreement / investigate and], Code is an alphabetic character with high operating frequency (step 5.4). However, Code is a shifted JIS code and assumes 8240h-82FFh to be the character code field where operating frequency is high here. When judged with operating frequency being high here, ON is set to Table_A [Code8240h] which shows whether the alphabetic character expressed with Code is registered, and processing (step 5.6) is ended. When judged with operating frequency being low at step 5.4, Code is added as an entry into Table_B of the alphabetic character registration managed table Table [i] (step 5.7), and processing is ended. It is necessary to search it for a vacant entry as usual in the case of the addition of the entry in step 5.7. Also in this case, it searches at a high speed using a Hash Function etc., and an entry is added. In addition, in using a Hash Function at the time of the addition of an entry, also in the processing which judges whether it is finishing [the download in step 3.7], it searches for Table_B using a Hash Function.

[0029] When the data of Table [i] differ from each data of printing data at step 5.3 of Table [i], it shifts to the processing about the following alphabetic character registration managed table

(step 5.5). and when the match of each data does not exist even to the last alphabetic character registration managed table An alphabetic character registration managed table is newly created (step 5.8). (step 5.11) [whether ON is set to registration flag Table_A [Code-8240h] of an alphabetic character expressed with Code of the alphabetic character registration managed table like processing of step 5.4, step 5.6, and step 5.7, and] Code is added as an entry into Table_B (step 5.10), and processing is ended.

[0030] Here, in processing of step 5.7, the situation which cannot download an alphabetic character image data newly may occur. This is the case where Table_B is already filled with the downloaded character code. In this case, a printer driver processes by changing printing by download to other printing approaches (for example, image drawing), or downloads instead of one entry of Table_B. In the case of the latter, it is necessary to notify deletion of an alphabetic character image data to a printer, and to download a new alphabetic character image data.

However, about an alphabetic character with high operating frequency, management of download is made by Table_A and does not become the situation which is not downloadable.

[0031] thus, finishing [the alphabetic character image data / the download to a printer] about an alphabetic character with high operating frequency — it is — compared with the conventional method of storing all download character codes on memory, when setting the flag which shows things, since a flag can be uniquely found from a character code, the time amount which the set of a flag takes is short, and ends. Moreover, about an alphabetic character with high operating frequency, it is always downloadable.

[0032] In addition, also in the management of cache data at the time of storing the alphabetic character image data which once rasterized on memory in the font rasterizer which creates an alphabetic character image data from outline data etc. (cash advance), this invention is applicable on a computer.

[0033] Moreover, it is the scalable font defined by the outline data which a printer contains, and this invention can be applied also in the management of cache data at the time of carrying out the cache of the alphabetic character image data which once performed the scaling on memory.

[0034] Moreover, about a character code, this invention is applicable also not only in the shifted JIS code shown in the example but JIS, UNICODE, the character code expressed by other two or more bytes.

[0035] Moreover, about language, if not only Japanese shown in the example but a character code is the language expressed by two or more bytes, this invention is applicable.

[0036] Moreover, even if the download data used as an administration object are outline data of the alphabetic character expressed not only by the alphabetic character image data shown in the example but by Beziers, a spline, etc., they can apply this invention.

[0037]

[Effect of the Invention] As explained above, according to this invention, the effectiveness which always enables download of a font with high operating frequency is done so.

[0038] Moreover, as explained above, in order to judge whether it is finishing [download] according to this invention, the effectiveness which enables compaction of the time amount to require is done so.

[0039] Moreover, as explained above, according to this invention, the effectiveness that memory size which a registered managed table needs can be lessened is done so.

[Translation done.]

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

TECHNICAL FIELD

[Industrial Application] This invention is a storage which memorizes the font download manager performed with an information processor and the information processors concerned, such as a host computer which downloads a font to an airline printer.

[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

PRIOR ART

[Description of the Prior Art] The environment of WYSIWYG (What You See IsWhat You Get) which prints to a printer by the same design as the alphabetic character displayed on a display is realized by having the device which generates an alphabetic character image data in a host computer, and creating and transmitting the alphabetic character image data doubled with each resolution to a display and a printer in recent years. It is necessary to transmit the image data of an alphabetic character to a printer in such a printing environment. In case it prints in this environment conventionally, the method (character image drawing) of transmitting an alphabetic character image data for every single character or the alphabetic character image data transmitted once was memorized in the printer, and the method (character image download) of reusing the image data memorized when the same alphabetic character was printed has been used.

[0003] The management method which identifies the alphabetic character which uses as one character set the alphabetic character group which makes the same the attributes (a font name, a character size, cell size, character decoration, etc.) of an alphabetic character by the approach of printing by this character image download, and is contained in that character set by the character code is common. In this case, one alphabetic character image data will be expressed in the combination of a character set and a character code.

[0004] In order to manage alphabetic character printing by said download from a host computer side, the data of being finishing [the alphabetic character image data of an alphabetic character which is going to print / download] already are needed. For this reason, the data in which it is shown for every character set whether it is finishing [download of each alphabetic character] are managed. The data area which stores the data in which it is shown hereafter whether it is finishing [this download] will be called a registered character code managed table. At least 1 bit which usually shows [download ending or] whether that is right to each character code is required for this data. Or although there is also a method of downloading beforehand all the alphabetic characters contained in the character set, since the amount of data becomes huge in an alphabetic coded character set as which a character code is expressed by two or more bytes, it is not realistic. Therefore, said management is needed in order to perform Japanese alphabetic character download.

[0005] Although the management memory [finishing / download] which is needed on a host in order to realize this is at least 1 bit per each alphabetic character, when it prepares the memory (from thousands to 10,000 [The case of Japanese] thousands of bits) of the number of bits corresponding to all character codes for every character set, it will press the memory by the side of a host. Since there is generally a limitation in the RAM capacity for memorizing on the printer in fact when printing by download, the number of alphabetic characters which can usually be registered is restricted. For this reason, the number of the maximum registered alphabetic characters was conventionally prepared for every character set, the memory which stores the character code for several of those alphabetic character minutes was prepared, and downloaded character code management has been performed by the approach of keeping as a list the character code downloaded there. and a character image downloads -- ** -- finishing [it is alike and / download] -- it is -- the data which recorded the data in which things are shown on a

registered character code managed table, stopped printing by download, changed and outputted to other printing approaches (for example, image drawing) when the condition that it is not downloadable during printing is reached, or have already downloaded deleted, and printing has been conventionally performed by the approach of downloading to the vacant part.

[Translation done.]

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

EFFECT OF THE INVENTION

[Effect of the Invention] As explained above, according to this invention, the effectiveness which always enables download of a font with high operating frequency is done so.

[0038] Moreover, as explained above, in order to judge whether it is finishing [download] according to this invention, the effectiveness which enables compaction of the time amount to require is done so.

[0039] Moreover, as explained above, according to this invention, the effectiveness that memory size which a registered managed table needs can be lessened is done so.

[Translation done.]

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

TECHNICAL PROBLEM

[Problem(s) to be Solved by the Invention] However, by this conventional approach, after reaching the upper limit of the number of alphabetic characters which can be registered, the problem that the image data of an alphabetic character with high operating frequency was not newly downloadable might occur.

[0007] Moreover, in order to manage a character code [finishing / download] by the list, even if it was the case of an alphabetic character with high operating frequency, the problem of taking time amount was in retrieval on the occasion of the judgment of being finishing [download], and record.

[0008] Moreover, when an alphabetic character registration managed table was created so that the character code downloadable for every character set for several maximum alphabetic character minutes can be stored, the memory size on the host who uses it became large, and there was a problem that it cannot secure.

[0009] This invention was made in order to solve these troubles, and it aims at always enabling download of an alphabetic character with high operating frequency.

[0010] Moreover, it aims at shortening the time amount required in order to be related with an alphabetic character with high operating frequency and to judge or record whether it is finishing [download].

[0011] Furthermore, it aims at lessening memory size which a registered character code managed table needs.

[Translation done.]

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

MEANS

[Means for Solving the Problem] It has a decision means the information processor concerning this invention is the information processor which downloads a font to an airline printer, and judge the operating frequency of the font which should print with said airline printer, and a selection means will choose the 2nd managed table if the 1st managed table will choose if it judges that said font which should be printed is a font with high operating frequency with said decision means, and said font which should print judges that it is a font with low operating frequency with said decision means.

[0013] Moreover, the storage concerning this invention is a storage which memorizes the font download manager performed with the information processor which downloads a font to an airline printer. The decision process which judges the operating frequency of the font which should be printed with said airline printer, If it judges that said font which should be printed is a font with high operating frequency at said decision process, the 1st managed table will be chosen. If it judges that said font which should be printed is a font with low operating frequency at said decision process, a font download manager equipped with the selection process which chooses the 2nd managed table will be memorized.

[Translation done.]

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.*** shows the word which can not be translated.

3.In the drawings, any words are not translated.

OPERATION

[Function] If it judges that the font which should choose the 1st managed table and should be printed is a font with low operating frequency if it judges that the font which should be printed in this invention by making it the above-mentioned configuration is a font with high operating frequency, the 2nd managed table will be chosen.

[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.

2. **** shows the word which can not be translated.

3. In the drawings, any words are not translated.

EXAMPLE

[Example] Next, the suitable example of this invention is explained to a detail with reference to a drawing.

[0016] Drawing 1 is the block diagram of the print control unit which is one example of this invention. In addition, if the function of this invention is performed, even if it is the system by which processing is performed through networks, such as LAN, even if it is the system which consists of two or more devices even if it is the device of a simple substance, it cannot be overemphasized that this invention is applicable.

[0017] In drawing, 3000 is a host computer, it has CPU1 which performs the document processing system in which the graphic form, the image, the alphabetic character, the table (a spreadsheet etc. is included), etc. were intermingled based on the document processing system program memorized by ROM for a program or external memory 11 of ROM3, and CPU1 controls each device connected to a system bus 4 in the gross.

[0018] Moreover, to ROM for a program or external memory 11 of this ROM3, the operating system program (henceforth, OS) which is a control program of CPU1 is memorized, the font data used for ROM for fonts or external memory 11 of ROM3 in the case of the above-mentioned document processing system is memorized to it, and ROM for data of ROM3 memorizes to it the various data used in case the above-mentioned document processing system etc. is performed. 2 is RAM and functions as the main memory of CPU1, a work area, etc. 5 is a keyboard controller (KBC) and controls the key input from a keyboard 9 or a non-illustrated pointing device. 6 is a CRT controller (CRTC) and controls the display of CRT display (CRT) 10. 7 is a disk controller (DKC) and controls access with the external memory 11 which memorizes a boot program, various kinds of applications, font data, a user file, an edit file, an alphabetic character image-data generator (following font writer riser), a printer control command generator (following printer driver), etc., such as a hard disk (HD) and a floppy disk (FD). 8 is a printer controller (PRTC), and it connects with a printer 1500 through the predetermined bidirection interface (interface) 21, and it performs communications control processing with a printer 1500. In addition, CPU1 performs expansion (writer rise) processing of the outline font to the display information RAM set up for example, on RAM2, and makes WYSIWYG on CRT10 possible. Moreover, CPU1 opens the various windows registered based on the command directed by the mouse cursor which is not illustrated on CRT10, and performs various data processing.

[0019] In a printer 1500, 12 is Printer CPU and outputs the picture signal as a print-out to the printing section (printer engine) 17 connected to a system bus 15 based on the control program memorized by external memory 14, such as a control program memorized by ROM for a program of ROM13. Moreover, the control program of CPU12 etc. is memorized for the program ROM of this ROM13. The font data used in case the above-mentioned print-out is generated to ROM for fonts of ROM13 was memorized, and in being the printer no external memory 14, such as a hard disk, is [printer] in ROM for data of ROM13, it has memorized the information used on a host computer. The communications processing with a host computer of CPU12 has become possible through the input section 18, and the host computer 3000 constitutes the information in a printer etc. possible [a notice]. 19 is RAM which functions as the main memory of CPU12, a

work area, etc., and it is constituted so that memory space can be extended with the option RAM connected to the extension port which is not illustrated. In addition, RAM19 is used for a print-out expansion field, an environmental data storage field, NVRAM, etc. The external memory 14 mentioned above, such as a hard disk (HD) and an IC card, has access controlled by the memory controller (MC) 20. It connects as an option and external memory 14 memorizes font data, an emulation program, form data, etc. Moreover, a switch, an LED drop, etc. for actuation are allotted with the control panel which 1501 mentioned above.

[0020] Moreover, the external memory mentioned above may be constituted so that not only one piece but two or more external memory which stored the program which interprets the printer control language with which it has at least one or more pieces, and an option font card differs from a language system in addition to an internal-organs font can be connected. Furthermore, it has NVRAM which is not illustrated and you may make it memorize the printer mode setting information from a control panel 1501.

[0021] using the flow chart which shows hereafter the control program performed by CPU1, it is alike in detail and one example of this invention is described.

[0022] Drawing 2 is a flow chart which shows the general procedure in the printer driver which downloads an alphabetic character image data. First, the outline of the procedure shown by drawing 2 is described. At step 2.1, printing data are received from the operating system (OS) or application on the host computer which has controlled the printer driver. At step 2.2, it judges whether it is already registered to a printer in the image data of the alphabetic character shown by the printing data acquired at step 2.1. When judged with it being registered, it shifts to the processing (step 2.7) which generates the instruction which prints the registered alphabetic character. When judged with un-registering, an alphabetic character image data is acquired from Font ROM based on the printing data obtained at step 2.1 (step 2.3). At step 2.4, the printer command which registers into a printer the character image acquired at step 2.3 is generated, and it transmits to (step 2.4) and a printer (step 2.5). At step 2.6, the flag which shows that it is already registered to a printer in the alphabetic character shown by printing data is set. At step 2.7, the command which prints the alphabetic character already registered to the printer is generated, and it transmits to a printer (step 2.8).

[0023] It is the parts of the judgment processing in step 2.2, and the set of the registered flag in step 2.6 that the alphabetic character image-data management method which carries out this invention is enforced in such procedure of a series of.

[0024] Next, the part in which this invention is carried out is explained in detail. The flow chart which shows the procedure which judges whether it is registered in step 2.2 of drawing 2 is shown in drawing 3. In this processing, the alphabetic character registration managed table managed by the printer driver which carries out this invention is used. Here, although a font environment which an alphabetic character decides by Code, FaceName, Style, and Point is assumed, sufficient data to specify an alphabetic character according to an environment should just be contained. And FaceName, Style, and all Point(s) manage a match as one character set, and the example which uses one alphabetic character registration managed table to the alphabetic character belonging to the same character set is shown here. One alphabetic character registration managed table is constituted from following contents which have some data about a character set besides a registered character code managed table by RAM2.

[0025] Alphabetic-character registration managed table: FaceName Point Style Table_A Table_B - (data 1)

FaceName: Font face name Point:character-size Style: Initialize the variable used by this processing at the registered character code managed table step 3.1 for alphabetic characters with the low registered character code managed table Table_B:use frequency for alphabetic characters with high character decoration data Table_A:use frequency. At step 3.2, it investigates whether the last alphabetic character registration managed table was reached. At step 3.3, it judges [whether it is in agreement with FaceName of the printing data which all of FaceName of the alphabetic character registration managed table Table [i], Style, and Point acquired at step 2.1, Style, and Point, and] whether in investigating and being in agreement, the character code shows the alphabetic character with high operating frequency (step 3.4).

However, Code is a shifted JIS code and assumes 8240h–82FFh to be the character code field where operating frequency is high here. When judged with operating frequency being high here, Table_A [Code 8240h] is investigated (step 3.6), in ON, it judges with it being registered, and processing is ended. In OFF, it judges with un-registering, and processing is ended in it. In investigating whether Table_B [x] which makes Code a value exists in Table_B when judged with operating frequency being low at step 3.4 (step 3.7) and existing, it judges with it being registered and ends processing. In not existing, it judges with un-registering and ends processing. When the data of Table [i] differ from each data of printing data at step 3.3, it shifts to the processing about the following alphabetic character registration managed table (step 3.5). And when the match of each data does not exist even to the last alphabetic character registration managed table, it judges with un-registering (step 3.8).

[0026] Here, it explains in more detail about Table_A and Table_B. It is shown whether these tables already have the registered image data of each alphabetic character to a printer. The contents are shown in drawing 4. The table adds the downloaded character code to the list, and serves as partial Table_A which expresses registered and un-registering by 1 bit to each character code from registered and partial Table_B which carries out a non-registered expression. Table_A is a part currently assigned 1 bit at a time to the alphabetic character with high operating frequency. Here, 192 characters which the character code of 81FFh(s) followed from 8140h containing a full-size alphanumeric hiragana in the example of a shifted JIS code are assigned to Table_A. Generally, if it is a Japanese document, the operating frequency of a hiragana is high and alphabetic character kinds, such as a hiragana, are assigned to the character code field which continued by many character coding schemes. Therefore, if a character coding scheme exists in the character code field to which the alphabetic character with high operating frequency continued, it can apply not only Shift JIS but this invention. Moreover, although the hiragana was mentioned as the example, it cannot be overemphasized that it can carry out also to alphabetic character kinds, such as katakana and a notation (punctuation etc. is included). Moreover, it is also possible to assign a single byte code to the field of Table_A and to assign 2 bytes (or more than it) of code to the field of Table_B. Table_B is a field for recording having downloaded the alphabetic character image data of the character code which does not belong to Table_A, for example, if it is used for a Japanese 2-byte code, it will initialize the field of Table_B with the suitable value (for example, 0) to the coding scheme, and it creates a downloaded character code list by adding the character code which downloaded to the part as for which Table_B was vacant. In addition, the size of this field doubles the number of alphabetic characters managed by Table_A and Table_B, and it is made to become a part for an alphabetic character downloadable for every character set.

[0027] Thus, compared with the conventional method of storing all download character codes on memory, when the alphabetic character image data investigates the data (flag) in which it is shown to a printer whether it is finishing [download] about an alphabetic character with high operating frequency by creating an alphabetic character registration managed table, since a flag can be uniquely found from a character code, the time amount which a judgment takes is short and ends. Moreover, since the flag of an alphabetic character with high operating frequency is 1 bit each, it can lessen memory usage of the whole used for download management compared with the case where the memory of several downloadable alphabetic character minute character code storing is used. In addition, about Table_B, it is necessary to search for a character code as usual. A character code is judged a numeric value and this part is searched at a high speed with the application of the Hash Function which is well-known technique. In addition, if a Hash Function is used at the time of retrieval, also in case it records download ending, it is necessary to store a character code in Table_B using a Hash Function.

[0028] At step 2.6 of drawing 2, the flag of the alphabetic character registered into the alphabetic character registration managed table to the printer at step 2.5 is set. Drawing 5 shows the flow chart of this processing. A variable required of this processing is initialized at step 5.1. At step 5.2, it investigates whether the last alphabetic character registration managed table was reached. At step 5.3, it investigates whether in when [which are in agreement with FaceName of the printing data which all of FaceName of the alphabetic character registration managed table

Table [i], Style, and Point acquired at step 2.1, Style, and Point / or or when in agreement / investigate and], Code is an alphabetic character with high operating frequency (step 5.4). However, Code is a shifted JIS code and assumes 8240h–82FFh to be the character code field where operating frequency is high here. When judged with operating frequency being high here, ON is set to Table_A [Code8240h] which shows whether the alphabetic character expressed with Code is registered, and processing (step 5.6) is ended. When judged with operating frequency being low at step 5.4, Code is added as an entry into Table_B of the alphabetic character registration managed table Table [i] (step 5.7), and processing is ended. It is necessary to search it for a vacant entry as usual in the case of the addition of the entry in step 5.7. Also in this case, it searches at a high speed using a Hash Function etc., and an entry is added. In addition, in using a Hash Function at the time of the addition of an entry, also in the processing which judges whether it is finishing [the download in step 3.7], it searches for Table_B using a Hash Function.

[0029] When the data of Table [i] differ from each data of printing data at step 5.3 of Table [i], it shifts to the processing about the following alphabetic character registration managed table (step 5.5). and when the match of each data does not exist even to the last alphabetic character registration managed table An alphabetic character registration managed table is newly created (step 5.8). (step 5.11) [whether ON is set to registration flag Table_A [Code–8240h] of an alphabetic character expressed with Code of the alphabetic character registration managed table like processing of step 5.4, step 5.6, and step 5.7, and] Code is added as an entry into Table_B (step 5.10), and processing is ended.

[0030] Here, in processing of step 5.7, the situation which cannot download an alphabetic character image data newly may occur. This is the case where Table_B is already filled with the downloaded character code. In this case, a printer driver processes by changing printing by download to other printing approaches (for example, image drawing), or downloads instead of one entry of Table_B. In the case of the latter, it is necessary to notify deletion of an alphabetic character image data to a printer, and to download a new alphabetic character image data. However, about an alphabetic character with high operating frequency, management of download is made by Table_A and does not become the situation which is not downloadable.

[0031] thus, finishing [the alphabetic character image data / the download to a printer] about an alphabetic character with high operating frequency — it is — compared with the conventional method of storing all download character codes on memory, when setting the flag which shows things, since a flag can be uniquely found from a character code, the time amount which the set of a flag takes is short, and ends. Moreover, about an alphabetic character with high operating frequency, it is always downloadable.

[0032] In addition, also in the management of cache data at the time of storing the alphabetic character image data which once rasterized on memory in the font rasterizer which creates an alphabetic character image data from outline data etc. (cash advance), this invention is applicable on a computer.

[0033] Moreover, it is the scalable font defined by the outline data which a printer contains, and this invention can be applied also in the management of cache data at the time of carrying out the cache of the alphabetic character image data which once performed the scaling on memory.

[0034] Moreover, about a character code, this invention is applicable also not only in the shifted JIS code shown in the example but JIS, UNICODE, the character code expressed by other two or more bytes.

[0035] Moreover, about language, if not only Japanese shown in the example but a character code is the language expressed by two or more bytes, this invention is applicable.

[0036] Moreover, even if the download data used as an administration object are outline data of the alphabetic character expressed not only by the alphabetic character image data shown in the example but by Beziers, a spline, etc., they can apply this invention.

*** NOTICES ***

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is a block diagram explaining the configuration of the information processing system which uses the alphabetic character image-data download management method which carries out this invention.

[Drawing 2] It is the flow chart which shows the procedure in the printer driver which performs character image download.

[Drawing 3] It is the flow chart which shows the procedure which judges whether it is registered in the character image download management method which carries out this invention.

[Drawing 4] It is drawing explaining the registered code table which records that the image data of each alphabetic character is download ending.

[Drawing 5] It is the flow chart which shows the procedure which records the registered thing in the character image download management method which carries out this invention.

[Description of Notations]

1 CPU

2 RAM

3 ROM

4 System Bus

12 CPU

13 ROM

19 RAM

1500 Printer

3000 Host Computer

[Translation done.]

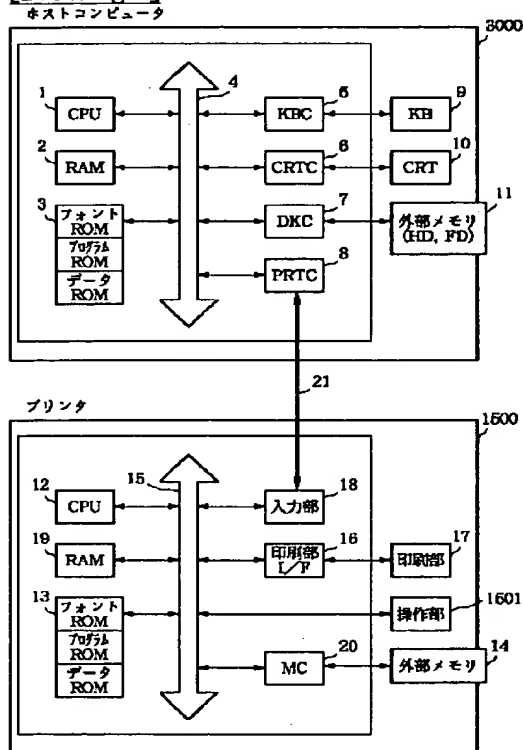
* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

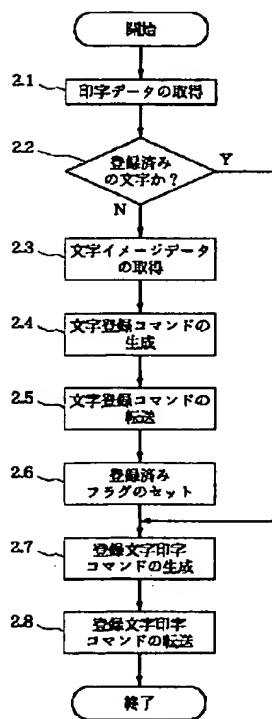
- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DRAWINGS

[Drawing 1]



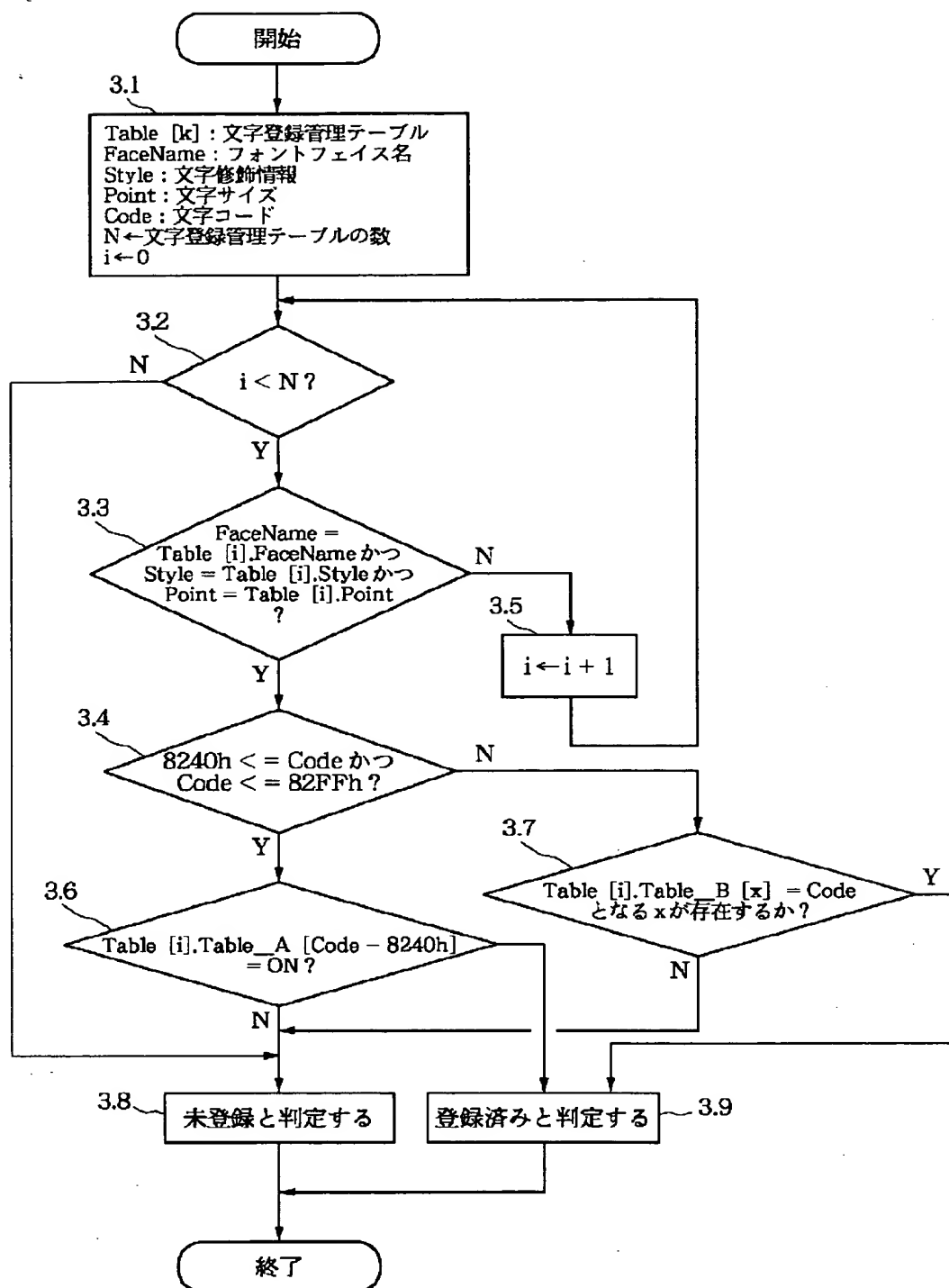
[Drawing 2]



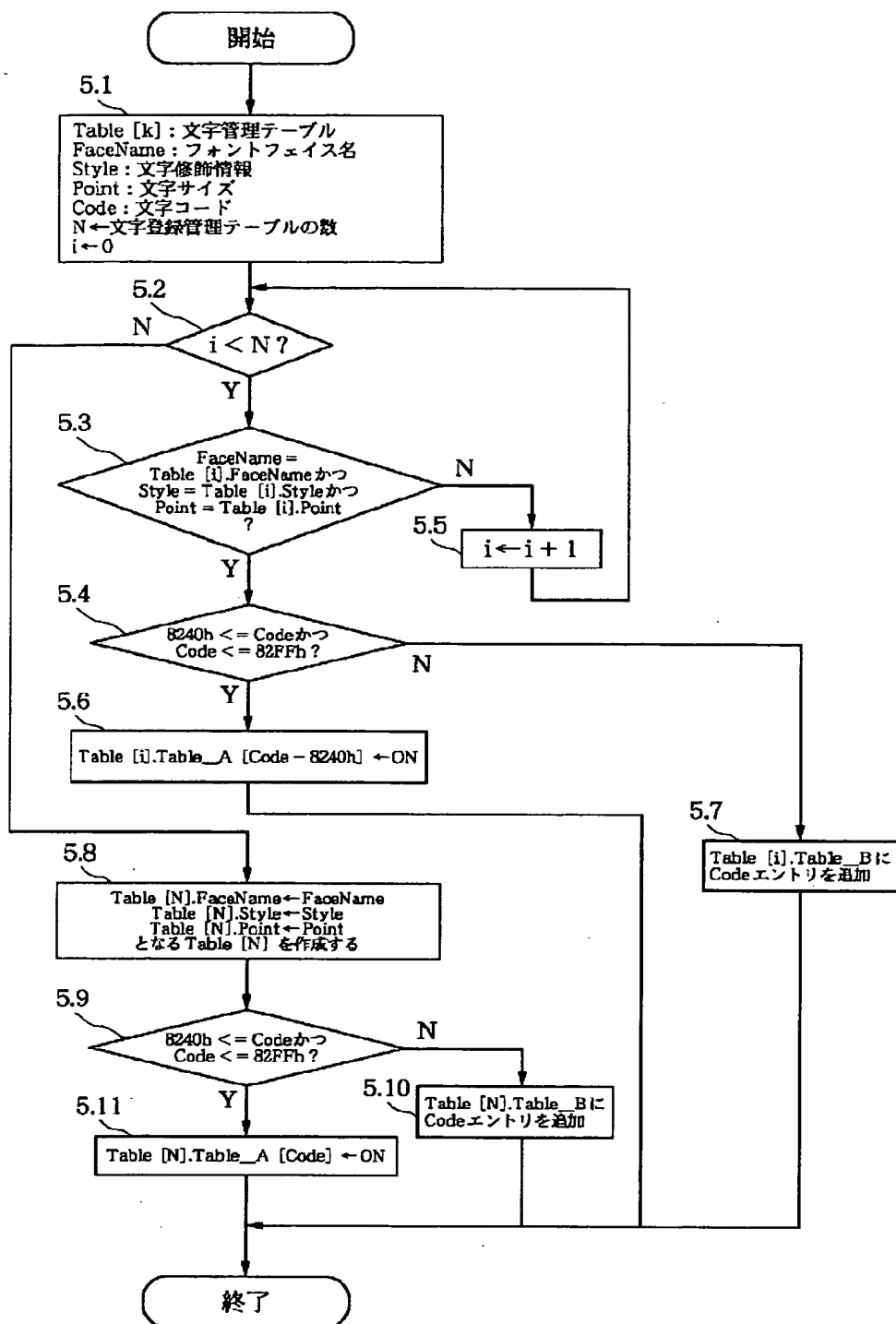
[Drawing 4]

| | |
|---------|--------------------------------|
| Table_A | 8240h - 82FFhの文字コードに対応した192ビット |
| Table_B | Table_A以外の文字コードを格納するエリア |

[Drawing 3]



[Drawing 5]



[Translation done.]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. *** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CORRECTION OR AMENDMENT

[Kind of official gazette] Printing of amendment by the convention of 2 of Article 17 of Patent Law

[Section partition] The 3rd partition of the 6th section

[Publication date] September 13, Heisei 14 (2002. 9.13)

[Publication No.] JP,9-16355,A

[Date of Publication] January 17, Heisei 9 (1997. 1.17)

[Annual volume number] Open patent official report 9-164

[Application number] Japanese Patent Application No. 7-160722

[The 7th edition of International Patent Classification]

G06F 3/12
B41J 2/485
5/44

[F]

G06F 3/12 G
B41J 5/44
3/12 C

[Procedure revision]

[Filing Date] June 27, Heisei 14 (2002. 6.27)

[Procedure amendment 1]

[Document to be Amended] Specification

[Item(s) to be Amended] Claim

[Method of Amendment] Modification

[Proposed Amendment]

[Claim(s)]

[Claim 1] A storage means to memorize the 2nd table for distinguishing whether the font data corresponding to an alphabetic character with low 1st table for distinguishing whether the font data corresponding to an alphabetic character with high operating frequency is registered into the airline printer and operating frequency is registered into the airline printer.

A distinction means to distinguish whether the font data corresponding to the alphabetic character which should be printed is registered into the airline printer based on said 1st table or said 2nd table.

The information processor which carries out [having a transmitting means transmit the command which registers the font data corresponding to the alphabetic character which should be printed, and the command which print the font data corresponding to the alphabetic character which should print registered to an airline printer, when the command which prints the font data corresponding to the alphabetic character which should be printed registered when were registered by said distinction means and it is distinguished transmits to an airline printer, it was

not registered by said distinction means and it is distinguished, and] as the description.

[Claim 2] Said 1st table is expressed by 1 bit in the registered no corresponding to the character code of an alphabetic character with high operating frequency,

Said 2nd table is an information processor according to claim 1 characterized by expressing whether it is registered by storing the character code of the alphabetic character registered.

[Claim 3] It distinguishes whether said distinction means is registered according to the condition of the bit of the location corresponding to the character code of the alphabetic character which should print said 1st table,

Said distinction means is an information processor according to claim 1 characterized by distinguishing whether it is registered or not according to whether the character code corresponding to the alphabetic character which should be printed on said 2nd table is stored.

[Claim 4] Said font data is an information processor according to claim 1 characterized by including an image data.

[Claim 5] A judgment means to judge whether the character code of the alphabetic character which should be printed is below a predetermined value,

The information processor according to claim 1 characterized by having a selection means to choose said 1st table when judged with it being below a predetermined value by said judgment means, and to choose said 2nd table when judged with it not being below a predetermined value by said judgment means.

[Claim 6] The table containing said the 1st table and said 2nd table is an information processor according to claim 1 characterized by more than one existing corresponding to a typeface.

[Claim 7] It is the information processing approach of using the 2nd table for distinguishing whether the font data corresponding to an alphabetic character with low 1st table for distinguishing whether the font data corresponding to an alphabetic character with high operating frequency being registered into the airline printer and operating frequency being registered into the airline printer,

The distinction step which distinguishes whether the font data corresponding to the alphabetic character which should be printed is registered into the airline printer based on said 1st table or said 2nd table,

When were registered by said distinction step and it is distinguished, the command which prints the font data corresponding to the alphabetic character which should be printed registered is transmitted to an airline printer. When were not registered by said distinction step and it is distinguished, The information processing approach characterized by having the transmitting step which transmits the command which registers the font data corresponding to the alphabetic character which should be printed, and the command which prints the font data corresponding to the alphabetic character which should be printed registered to an airline printer.

[Claim 8] Said 1st table is expressed by 1 bit in the registered no corresponding to the character code of an alphabetic character with high operating frequency,

Said 2nd table is the information processing approach according to claim 7 characterized by expressing whether it is registered by storing the character code of the alphabetic character registered.

[Claim 9] It distinguishes whether said distinction step is registered according to the condition of the bit of the location corresponding to the character code of the alphabetic character which should print said 1st table,

Said distinction step is the information processing approach according to claim 7 characterized by distinguishing whether it is registered or not according to whether the character code corresponding to the alphabetic character which should be printed on said 2nd table is stored.

[Claim 10] Said font data is the information processing approach according to claim 7 characterized by including an image data.

[Claim 11] The judgment step which judges whether the character code of the alphabetic character which should be printed is below a predetermined value,

When judged with it being below a predetermined value by said judgment step, said 1st table is chosen. The information processing approach according to claim 7 characterized by having the selection step which chooses said 2nd table when judged with it not being below a predetermined

value by said judgment step.

[Claim 12] The table containing said the 1st table and said 2nd table is the information processing approach according to claim 7 characterized by more than one existing corresponding to a typeface.

[Procedure amendment 2]

[Document to be Amended] Specification

[Item(s) to be Amended] 0012

[Method of Amendment] Modification

[Proposed Amendment]

[0012]

[Means for Solving the Problem] In order to attain the above-mentioned purpose the information processor of this invention A storage means to memorize the 2nd table for distinguishing whether the font data corresponding to an alphabetic character with low 1st table for distinguishing whether the font data corresponding to an alphabetic character with high operating frequency is registered into the airline printer and operating frequency is registered into the airline printer, A distinction means to distinguish whether the font data corresponding to the alphabetic character which should be printed is registered into the airline printer based on said 1st table or said 2nd table, When were registered by said distinction means and it is distinguished, the command which prints the font data corresponding to the alphabetic character which should be printed registered is transmitted to an airline printer. When were not registered by said distinction means and it is distinguished, it is characterized by having a transmitting means to transmit the command which registers the font data corresponding to the alphabetic character which should be printed, and the command which prints the font data corresponding to the alphabetic character which should be printed registered to an airline printer.

[Procedure amendment 3]

[Document to be Amended] Specification

[Item(s) to be Amended] 0013

[Method of Amendment] Modification

[Proposed Amendment]

[0013] Moreover, the information processing approach of this invention It is the information processing approach of using the 2nd table for distinguishing whether the font data corresponding to an alphabetic character with low 1st table for distinguishing whether the font data corresponding to an alphabetic character with high operating frequency being registered into the airline printer and operating frequency being registered into the airline printer. The distinction step which distinguishes whether the font data corresponding to the alphabetic character which should be printed is registered into the airline printer based on said 1st table or said 2nd table, When were registered by said distinction step and it is distinguished, the command which prints the font data corresponding to the alphabetic character which should be printed registered is transmitted to an airline printer. When were not registered by said distinction step and it is distinguished, It is characterized by having the transmitting step which transmits the command which registers the font data corresponding to the alphabetic character which should be printed, and the command which prints the font data corresponding to the alphabetic character which should be printed registered to an airline printer.

[Procedure amendment 4]

[Document to be Amended] Specification

[Item(s) to be Amended] 0014

[Method of Amendment] Deletion

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-16355

(43) 公開日 平成9年(1997)1月17日

| (51) Int.Cl. ⁸ | 識別記号 | 庁内整理番号 | F I | 技術表示箇所 |
|---------------------------|------|--------|--------------|--------|
| G 0 6 F 3/12 | | | G 0 6 F 3/12 | G |
| B 4 1 J 2/485 | | | B 4 1 J 5/44 | |
| 5/44 | | | 3/12 | C |

審査請求 未請求 請求項の数11 O L (全 9 頁)

(21) 出願番号 特願平7-160722

(22) 出願日 平成7年(1995)6月27日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 中桐 孝治

東京都大田区下丸子3丁目30番2号キヤノ
ン株式会社内

(74) 代理人 弁理士 丸島 儀一

(54) 【発明の名称】 情報処理装置及びフォントダウンロード管理プログラムを記憶する記憶媒体

(57) 【要約】

【目的】 使用頻度の高いフォントのダウンロードを常に可能とし、ダウンロード済みか否かを判断する為に要する時間の短縮と登録済み管理テーブルが必要とするメモリサイズを少なくする。

【構成】 印刷すべきフォントが使用頻度が高いフォントであると判断すると第1管理テーブル（印刷装置で印刷すべき使用頻度が高いフォントが印刷装置にダウンロード済みであるか否かを示す情報を1ビットで管理するテーブル）を選択し、印刷すべきフォントが使用頻度が低いフォントであると判断すると第2管理テーブル（印刷装置で印刷すべき使用頻度が低いフォントが印刷装置にダウンロード済みであるか否かを示す情報を文字コード長のデータで管理するテーブル）を選択する。

Table_A

8240h - 82FFhの文字コードに対応した192ビット

Table_B

Table_A以外の文字コードを格納するエリア

【特許請求の範囲】

【請求項 1】 印刷装置へフォントをダウンロードする情報処理装置であって、
前記印刷装置で印刷すべきフォントの使用頻度を判断する判断手段と、
前記判断手段で印刷すべきフォントが使用頻度が高いフォントであると判断すると第 1 管理テーブルを選択し、
前記判断手段で前記印刷すべきフォントが使用頻度が低いフォントであると判断すると第 2 管理テーブルを選択する選択手段とを備えることを特徴とする情報処理装置。

【請求項 2】 前記判断手段は、単一の連続したフォントコード領域或は複数の連続したフォントコード領域を使用頻度の高いフォントコード領域としての判断基準とすることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 前記第 1 管理テーブルは、前記印刷装置で印刷すべき使用頻度が高いフォントが前記印刷装置にダウンロード済みであるか否かを示す情報を 1 ビットで管理することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 4】 前記第 2 管理テーブルは、前記印刷装置で印刷すべき使用頻度が低いフォントが前記印刷装置にダウンロード済みであるか否かを示す情報を文字コード長のデータで管理することを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】 前記選択手段で選択された第 1 管理テーブル或は第 2 管理テーブルを用いて、前記印刷すべきフォントが前記印刷装置にダウンロード済みであるか否か判断されることを特徴とする請求項 1 に記載の情報処理装置。

【請求項 6】 印刷装置へフォントをダウンロードする情報処理装置で実行されるフォントダウンロード管理プログラムを記憶する記憶媒体であって、
前記印刷装置で印刷すべきフォントの使用頻度を判断する判断工程と、

前記判断工程で前記印刷すべきフォントが使用頻度が高いフォントであると判断すると第 1 管理テーブルを選択し、前記判断工程で前記印刷すべきフォントが使用頻度が低いフォントであると判断すると第 2 管理テーブルを選択する選択工程とを備えるフォントダウンロード管理プログラムを記憶することを特徴とする記憶媒体。

【請求項 7】 前記判断工程は、単一の連続したフォントコード領域或は複数の連続したフォントコード領域を使用頻度の高いフォントコード領域としての判断基準とすることを特徴とする請求項 6 に記載の記憶媒体。

【請求項 8】 前記第 1 管理テーブルは、前記印刷装置で印刷すべき使用頻度が高いフォントが前記印刷装置にダウンロード済みであるか否かを示す情報を 1 ビットで管理することを特徴とする請求項 6 に記載の記憶媒体。

【請求項 9】 前記第 2 管理テーブルは、前記印刷装置

で印刷すべき使用頻度が低いフォントが前記印刷装置にダウンロード済みであるか否かを示す情報を文字コード長のデータで管理することを特徴とする請求項 6 に記載の記憶媒体。

【請求項 10】 前記選択工程で選択された第 1 管理テーブル或は第 2 管理テーブルを用いて、前記印刷すべきフォントが前記印刷装置にダウンロード済みであるか否か判断されることを特徴とする請求項 6 に記載の記憶媒体。

【請求項 11】 前記判断工程と前記選択工程を備えるフォントダウンロード管理プログラムは、前記情報処理装置に着脱可能な外部記憶装置であることを特徴とする請求項 6 に記載の記憶媒体。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、印刷装置へフォントをダウンロードするホストコンピュータ等の情報処理装置及び当該情報処理装置で実行されるフォントダウンロード管理プログラムを記憶する記憶媒体。

【0002】

【従来技術】近年、ホストコンピュータ内に文字イメージデータを生成する機構を持ち、ディスプレイ及びプリンタへ、それぞれの解像度に合わせた文字イメージデータを作成し、転送することにより、ディスプレイ上に表示される文字と同一のデザインでプリンタへ印字を行う WYSIWYG (What You See Is What You Get) という環境が実現されている。このような印字環境では、プリンタへ文字のイメージデータを転送する必要がある。従来、この環境で印字を行う際には、一文字毎に文字イメージデータを転送する方法 (文字イメージ描画)、あるいは一度転送した文字イメージデータをプリンタ内に記憶しておき、同一の文字を印字する場合に記憶しておいたイメージデータを再利用するといった方法 (文字イメージダウンロード) が用いられてきた。

【0003】この文字イメージダウンロードにより印字を行う方法では、文字の属性 (フォント名称、文字サイズ、セルサイズ、文字修飾等) を同じくする文字群を一つの文字セットとし、その文字セットに含まれる文字を文字コードによって識別する管理方法が一般的である。この場合、文字セットと文字コードの組み合わせで一つの文字イメージデータを表現することとなる。

【0004】前記ダウンロードによる文字印字をホストコンピュータ側から管理するためには、印字を行おうとする文字の文字イメージデータが既にダウンロード済みかどうかのデータが必要となる。このため、文字セットごとに各文字がダウンロード済みかどうかを示すデータを管理する。以下、このダウンロード済みかどうかを示すデータを格納するデータ領域を登録済み文字コード管理テーブルと呼ぶことにする。このデータは通常各文字

コードに対してダウンロード済みかそうでないかを示す最低 1 ビットが必要である。あるいはその文字セットに含まれるすべての文字をあらかじめダウンロードしておくという方法もあるが、文字コードが複数バイトで表現されるような文字コードセットではデータ量が膨大となるため現実的ではない。よって日本語の文字ダウンロードを行うためには前記管理が必要となる。

【0005】これを実現するためにホスト上に必要となるダウンロード済みの管理メモリは、各文字につき最低 1 ビットであるが、すべての文字コードに対応したビット数のメモリ（日本語の場合には数千から 1 万数千ビット）を文字セットごとに用意すると、ホスト側のメモリを圧迫することとなる。実際には、ダウンロードにより印字を行う場合、プリンタ上に記憶しておくための RAM 容量には一般に限界があるため、通常登録可能な文字数が制限される。このため、従来は文字セットごとに最大登録文字数を設け、その文字数分の文字コードを格納するメモリを用意し、そこにダウンロードした文字コードをリストとして保管するといった方法でダウンロード済みの文字コード管理が行われてきた。そして、文字イメージをダウンロードするごとにダウンロード済みであることを示すデータを登録済み文字コード管理テーブルに記録し、印刷中にダウンロードが行えない状態に達した場合には、ダウンロードによる印字をやめて他の印字方法（例えばイメージ描画）に切り替えて出力するか、あるいは既にダウンロードされているデータを削除して、その空いた部分にダウンロードを行うといった方法で従来印字が行われてきた。

【0006】

【発明が解決しようとしている課題】しかし、この従来の方法では、登録可能な文字数の上限に達した後は、使用頻度の高い文字のイメージデータを新規にダウンロード出来ないといった問題が発生することがあった。

【0007】また、ダウンロード済みの文字コードをリストで管理するため、使用頻度の高い文字の場合であってもダウンロード済みかどうかの判定および記録の際に、探索に時間がかかるという問題があった。

【0008】また、文字セットごとにダウンロード可能な最大文字数分の文字コードを格納できるように文字登録管理テーブルを作成した場合、その使用するホスト上のメモリサイズが大きくなり、確保できない場合も有りうるという問題があった。

【0009】本発明は、これらの問題点を解決するためになされたもので、使用頻度の高い文字のダウンロードを常に可能とすることを目的とする。

【0010】また、使用頻度の高い文字に関してダウンロード済みかどうかを判定、あるいは記録するために要する時間を短縮することを目的とする。

【0011】さらに、登録済み文字コード管理テーブルが必要とするメモリサイズを少なくすることを目的とす

る。

【0012】

【課題を解決するための手段】本発明に係る情報処理装置は、印刷装置へフォントをダウンロードする情報処理装置であって、前記印刷装置で印刷すべきフォントの使用頻度を判断する判断手段と、前記判断手段で前記印刷すべきフォントが使用頻度が高いフォントであると判断すると第 1 管理テーブルを選択し、前記判断手段で前記印刷すべきフォントが使用頻度が低いフォントであると判断すると第 2 管理テーブルを選択する選択手段とを備える。

【0013】また、本発明に係る記憶媒体は、印刷装置へフォントをダウンロードする情報処理装置で実行されるフォントダウンロード管理プログラムを記憶する記憶媒体であって、前記印刷装置で印刷すべきフォントの使用頻度を判断する判断工程と、前記判断工程で前記印刷すべきフォントが使用頻度が高いフォントであると判断すると第 1 管理テーブルを選択し、前記判断工程で前記印刷すべきフォントが使用頻度が低いフォントであると判断すると第 2 管理テーブルを選択する選択工程とを備えるフォントダウンロード管理プログラムを記憶する。

【0014】

【作用】上記構成にすることにより本発明においては、印刷すべきフォントが使用頻度が高いフォントであると判断すると第 1 管理テーブルを選択し、印刷すべきフォントが使用頻度が低いフォントであると判断すると第 2 管理テーブルを選択する。

【0015】

【実施例】次に、本発明の好適な実施例について図面を参照し詳細に説明する。

【0016】図 1 は本発明の一実施例である印刷制御装置のブロック図である。なお、本発明の機能が実行されるのであれば、単体の機器であっても、複数の機器からなるシステムであっても、LAN 等のネットワークを介して処理が行われるシステムであっても本発明を適用できることは言うまでもない。

【0017】図において、3000 はホストコンピュータで、ROM 3 のプログラム用 ROM あるいは外部メモリ 11 に記憶された文書処理プログラム等に基づいて図形、イメージ、文字、表（表計算等を含む）等が混在した文書処理を実行する CPU 1 を備え、システムバス 4 に接続される各デバイスを CPU 1 が総括的に制御する。

【0018】また、この ROM 3 のプログラム用 ROM あるいは外部メモリ 11 には、CPU 1 の制御プログラムであるオペレーティングシステムプログラム（以下 OS）等を記憶し、ROM 3 のフォント用 ROM あるいは外部メモリ 11 には上記文書処理の際に使用するフォントデータ等を記憶し、ROM 3 のデータ用 ROM は上記文書処理等を行う際に使用する各種データを記憶する。

2はRAMで、CPU1の主メモリ、ワークエリア等として機能する。5はキーボードコントローラ（KBC）で、キーボード9や不図示のポインティングデバイスからのキー入力を制御する。6はCRTコントローラ（CRTC）で、CRTディスプレイ（CRT）10の表示を制御する。7はディスクコントローラ（DKC）で、ブートプログラム、各種のアプリケーション、フォントデータ、ユーザファイル、編集ファイル、文字イメージデータ生成プログラム（以下フォントライタライザ）、プリンタ制御コマンド生成プログラム（以下プリンタドライバ）等を記憶するハードディスク（HD）、フロッピーディスク（FD）等の外部メモリ11とのアクセスを制御する。8はプリンタコントローラ（PRTC）で、所定の双方向性インターフェース（インターフェース）21を介してプリンタ1500に接続されて、プリンタ1500との通信制御処理を実行する。なお、CPU1は、例えばRAM2上に設定された表示情報RAMへのアウトラインフォントの展開（ライタライズ）処理を実行し、CRT10上でのWYSIWYGを可能としている。また、CPU1は、CRT10上の不図示のマウスカーソル等で指示されたコマンドに基づいて登録された種々のウインドウを開き、種々のデータ処理を実行する。

【0019】プリンタ1500において、12はプリンタCPUで、ROM13のプログラム用ROMに記憶された制御プログラム等あるいは外部メモリ14に記憶された制御プログラム等に基づいてシステムバス15に接続される印刷部（プリンタエンジン）17に出力情報としての画像信号を出力する。また、このROM13のプログラムROMには、CPU12の制御プログラム等を記憶する。ROM13のフォント用ROMには上記出力情報を生成する際に使用するフォントデータ等を記憶し、ROM13のデータ用ROMにはハードディスク等の外部メモリ14がないプリンタの場合には、ホストコンピュータ上で利用される情報等を記憶している。CPU12は入力部18を介してホストコンピュータとの通信処理が可能となっており、プリンタ内の情報等をホストコンピュータ3000に通知可能に構成されている。19はCPU12の主メモリ、ワークエリア等として機能するRAMで、図示しない増設ポートに接続されるオプションRAMによりメモリ容量を拡張することができるように構成されている。なお、RAM19は、出力情報展開領域、環境データ格納領域、NVRAM等に用いられる。前述したハードディスク（HD）、ICカード等の外部メモリ14は、メモリコントローラ（MC）20によりアクセスを制御される。外部メモリ14は、オプションとして接続され、フォントデータ、エミュレーションプログラム、フォームデータ等を記憶する。また、1501は前述した操作パネルで操作のためのスイッチおよびLED表示器等が配されている。

【0020】また、前述した外部メモリは1個に限らず、少なくとも1個以上備え、内臓フォントに加えてオプションフロントカード、言語系の異なるプリンタ制御言語を解釈するプログラムを格納した外部メモリを複数接続できるように構成されていてもよい。さらに、図示しないNVRAMを有し、操作パネル1501からのプリンタモード設定情報を記憶するようにしてもよい。

【0021】以下、CPU1によって実行される制御プログラムを示すフローチャートを用いて、本発明の一実施例について詳しく述べる。

【0022】図2は、文字イメージデータのダウンロードを行うプリンタドライバでの一般的な処理手順を示すフローチャートである。まず、図2で示される処理手順の概要を述べる。ステップ2.1では、プリンタドライバをコントロールしているホストコンピュータ上のオペレーティングシステム（OS）あるいはアプリケーションから印字データを受け取る。ステップ2.2では、ステップ2.1で取得した印字データで示される文字のイメージデータを既にプリンタへ登録済みかどうか判定する。登録済みと判定された場合には、登録した文字を印字する命令を生成する処理（ステップ2.7）へ移行する。未登録と判定された場合には、ステップ2.1で得た印字データを元に、文字イメージデータをフォントROMから取得する（ステップ2.3）。ステップ2.4ではステップ2.3で取得した文字イメージをプリンタに登録するプリンタコマンドを生成し、（ステップ2.4）、プリンタへ転送する（ステップ2.5）。ステップ2.6では印字データで示された文字を既にプリンタへ登録済みであることを示すフラグをセットしておく。ステップ2.7では既にプリンタへ登録されている文字を印字するコマンドを生成し、プリンタへ転送する（ステップ2.8）。

【0023】これらの一連の処理手順の中で、本発明を実施する文字イメージデータ管理方法が実施されるのはステップ2.2における判定処理、およびステップ2.6における登録済みフラグのセットの部分である。

【0024】次に、本発明が実施される部分を詳しく説明する。図2のステップ2.2における登録済みか否かを判定する処理手順を示すフローチャートを図3に示す。この処理では、本発明を実施するプリンタドライバによって管理される文字登録管理テーブルを使用する。ここでは、Code、FaceName、Style、Pointによって文字が確定する様なフォント環境を想定しているが、環境によって文字を特定するに十分なデータが含まれていればよい。そしてここでは、FaceName、Style、Pointすべてが一致するものを一つの文字セットとして管理し、同一の文字セットに属する文字に対して一つの文字登録管理テーブルを使用する例を示す。一つの文字登録管理テーブルは、登録済み文字コード管理テーブルの他に文字セットに関す

るいくつかのデータを持つ以下のような内容でRAM2に構成される。

【0025】文字登録管理テーブル：FaceName
Point Style Table__ATable
__B…（データ1）

FaceName：フォントフェイス名

Point：文字サイズ

Style：文字修飾データ

Table__A：使用頻度の高い文字用登録済み文字コード管理テーブル

Table__B：使用頻度の低い文字用登録済み文字コード管理テーブル

ステップ3. 1では、この処理で使用する変数の初期化を行う。ステップ3. 2では、最後の文字登録管理テーブルに達したか調べる。ステップ3. 3では、文字登録管理テーブルTable [i] のFaceName、Style、Pointがすべてステップ2. 1で取得した印字データのFaceName、Style、Pointと一致するか調べ、一致する場合には、その文字コードが使用頻度の高い文字を示しているかどうか判定する（ステップ3. 4）。但し、ここではCodeはシフトJISコードであり、8240h-82FFhを使用頻度の高い文字コード領域と仮定している。ここで使用頻度が高いと判定された場合には、Table__A [Code 8240h] を調べ（ステップ3. 6）、ONの場合には登録済みと判定して処理を終了する。OFFの場合には、未登録と判定して処理を終了する。ステップ3. 4で使用頻度が低いと判定された場合には、Table__Bの中にCodeを値とするTable__B [x] が存在するか調べ（ステップ3. 7）存在する場合には登録済みと判定して処理を終了する。存在しない場合には、未登録と判定して処理を終了する。ステップ3. 3でTable [i] のデータが印字データの各データと異なる場合には、次の文字登録管理テーブルに関する処理へ移行する（ステップ3. 5）。そして最後の文字登録管理テーブルまで各データの一致するものが存在しなかった場合には、未登録と判定する（ステップ3. 8）。

【0026】ここで、Table__AおよびTable__Bについて更に詳しく説明する。これらのテーブルは各文字のイメージデータがプリンタに対して既に登録済みであるか否かを示すものである。その内容を図4に示す。テーブルは、各文字コードに対して1ビットで登録済み、未登録を表現する部分Table__Aと、ダウンロードした文字コードをリストに加えていき、登録済み、未登録表現する部分Table__Bとからなる。Table__Aは、使用頻度の高い文字に対して1ビットずつ割り当てられている部分である。ここでは、シフトJISコードの例で、全角英数ひらがなを含む8140hから81FFhの文字コードの連続した192文字分

をTable__Aに割り当てている。一般に、日本語の文書であれば、ひらがなの使用頻度は高く、またひらがな等の文字種は多くの文字コード体系で連続した文字コード領域に割り当てられている。よって、文字コード体系は使用頻度の高い文字が連続した文字コード領域に存在するものであればシフトJISに限らず本発明を適用可能である。また、ひらがなを例に挙げたが、カタカナ、記号（句読点等を含む）等の文字種に対しても実施可能であることは言うまでもない。また、1バイトコードをTable__Aの領域に、2バイト（あるいはそれ以上）のコードをTable__Bの領域に割り当てることも可能である。Table__BはTable__Aに属さない文字コードの文字イメージデータをダウンロードしたことを記録するための領域であり、例えば日本語の2バイトコードに使用するのであれば、そのコード体系にふさわしい値（例えば0）でTable__Bの領域を初期化しておき、ダウンロードを行った文字コードをTable__Bの空いた部分に追加していくことによりダウンロード済みの文字コードリストを作成する。なお、この領域のサイズはTable__AとTable__Bで管理される文字数を合わせて文字セットごとにダウンロード可能な文字分となるようにしておく。

【0027】このように文字登録管理テーブルを作成することにより、使用頻度の高い文字に関しては、その文字イメージデータがプリンタへダウンロード済みかを示すデータ（フラグ）を調べる場合、すべてのダウンロード文字コードをメモリ上に格納しておく従来の方法と比べ、文字コードから一意にフラグが求まるので判定に要する時間が短くて済む。また、使用頻度の高い文字のフラグは各1ビットのため、ダウンロード可能な文字数分文字コード格納のメモリを使用する場合に比べて、ダウンロード管理に使用する全体のメモリ使用量を少なくすることができる。なお、Table__Bに関しては、従来通り、文字コードの探索を行う必要がある。この部分では、文字コードを数値と見立てて、公知の手法であるハッシュ関数等を適用して高速に探索を行う。なお、探索時にハッシュ関数を利用するのであれば、ダウンロード済み記録の際にもハッシュ関数を使用してTable__Bに文字コードを格納しておく必要がある。

【0028】図2のステップ2. 6では、文字登録管理テーブルにステップ2. 5でプリンタへ登録した文字のフラグをセットする。図5は、この処理のフローチャートを示す。ステップ5. 1では、この処理で必要な変数の初期化を行う。ステップ5. 2では、最後の文字登録管理テーブルに達したか調べる。ステップ5. 3では、文字登録管理テーブルTable [i] のFaceName、Style、Pointがすべてステップ2. 1で取得した印字データのFaceName、Style、Pointと一致するか調べ、一致する場合には、Codeが使用頻度の高い文字であるかを調べる（ステ

ップ5. 4)。但し、ここではCodeはシフトJISコードであり、8240h-82FFhを使用頻度の高い文字コード領域と仮定している。ここで使用頻度が高いと判定された場合には、Codeで表わされる文字が登録済みであるかどうかを示すTable__A [Code8240h]にONをセットして(ステップ5. 6)処理を終了する。ステップ5. 4で使用頻度が低いと判定された場合には、文字登録管理テーブルTable [i]のTable__Bの中にCodeをエン트리として追加し(ステップ5. 7)、処理を終了する。ステップ5. 7におけるエントリの追加の際には、従来と同様、空きエントリを探索する必要がある。この場合にも、ハッシュ関数等を利用して高速に探索を行い、エントリを追加する。なお、エントリの追加時にハッシュ関数を利用する場合には、ステップ3. 7におけるダウンロード済みかを判定する処理においても、ハッシュ関数を使用してTable__Bの探索を行う。

【0029】Table [i]のステップ5. 3でTable [i]のデータが印字データの各データと異なる場合には、次の文字登録管理テーブルに関する処理へ移行する(ステップ5. 5)。そして最後の文字登録管理テーブルまで各データの一致するものが存在しなかった場合には、新たに文字登録管理テーブルを作成し(ステップ5. 8)、ステップ5. 4、ステップ5. 6、ステップ5. 7の処理と同様にその文字登録管理テーブルのCodeで表わされる文字の登録フラグTable__A [Code-8240h]にONをセットするか(ステップ5. 11)、Table__Bの中にCodeをエン트리として追加し(ステップ5. 10)、処理を終了する。

【0030】ここで、ステップ5. 7の処理では、新規に文字イメージデータをダウンロード出来ない状況が発生する場合がある。これは、すでにTable__Bがダウンロード済み文字コードで満たされている場合である。この場合には、プリンタドライバはダウンロードによる印字を他の印字方法(例えばイメージ描画)へ切り替えて処理を行うか、Table__Bのいずれかのエントリの代わりにダウンロードを行う。後者の場合には、文字イメージデータの削除をプリンタへ通知し、新たな文字イメージデータをダウンロードする必要がある。ただし、使用頻度の高い文字に関しては、ダウンロードの管理がTable__Aでなされており、ダウンロード出来ない状況になることはない。

【0031】このように、使用頻度の高い文字に関しては、その文字イメージデータがプリンタへダウンロード済みであることを示すフラグをセットする場合、すべてのダウンロード文字コードをメモリ上に格納しておく従来の方法と比べ、文字コードから一意にフラグが求まるのでフラグのセットに要する時間が短くて済む。また、使用頻度の高い文字に関しては、ダウンロードが常に可

能である。

【0032】なお、コンピュータ上で文字イメージデータをアウトラインデータ等から作成するフォントラスライザにおいて、一旦ラスライズを行った文字イメージデータをメモリ上に格納(キャッシング)しておく際の、キャッシュデータの管理においても、本発明を適用可能である。

【0033】また、プリンタの内蔵するアウトラインデータ等で定義されたスケラブルなフォントで、一旦スケリングを行った文字イメージデータをメモリ上にキャッシュしておく際の、キャッシュデータの管理においても、本発明を適用可能である。

【0034】また、文字コードに関しては、実施例で示したシフトJISコードに限らず、JIS、UNICODEや、他の複数バイトで表現される文字コード等においても本発明を適用可能である。

【0035】また、言語に関しては、実施例で示した日本語に限らず、文字コードが複数バイトで表現される言語であれば本発明を適用可能である。

【0036】また、管理対象となるダウンロードデータは、実施例で示した文字イメージデータに限らず、ページやスプライン等で表現された文字のアウトラインデータであっても本発明を適用可能である。

【0037】

【発明の効果】以上説明したように本発明によれば、使用頻度の高いフォントのダウンロードを常に可能とする効果を奏する。

【0038】また、以上説明したように本発明によれば、ダウンロード済みか否かを判断する為に要する時間の短縮を可能とする効果を奏する。

【0039】また、以上説明したように本発明によれば、登録済管理テーブルが必要とするメモリサイズを少なくできるという効果を奏する。

【図面の簡単な説明】

【図1】本発明を実施する文字イメージデータダウンロード管理方法を使用する情報処理システムの構成を説明するブロック図である。

【図2】文字イメージダウンロードを行うプリンタドライバにおける処理手順を示すフローチャートである。

【図3】本発明を実施する文字イメージダウンロード管理方法における登録済みであるか否かを判定する処理手順を示すフローチャートである。

【図4】各文字のイメージデータがダウンロード済みであることを記録する登録済みコードテーブルを説明する図である。

【図5】本発明を実施する文字イメージダウンロード管理方法における登録済みであることを記録する処理手順を示すフローチャートである。

【符号の説明】

1 CPU

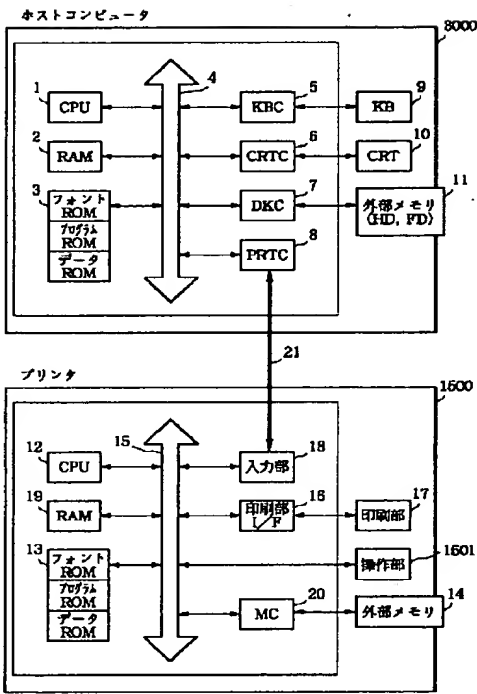
11

12

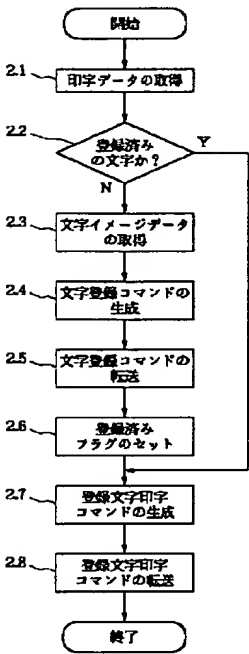
- 2 RAM
- 3 ROM
- 4 システムバス
- 1 2 CPU

- 13 ROM
- 19 RAM
- 1500 プリンタ
- 3000 ホストコンピュータ

【図 1】



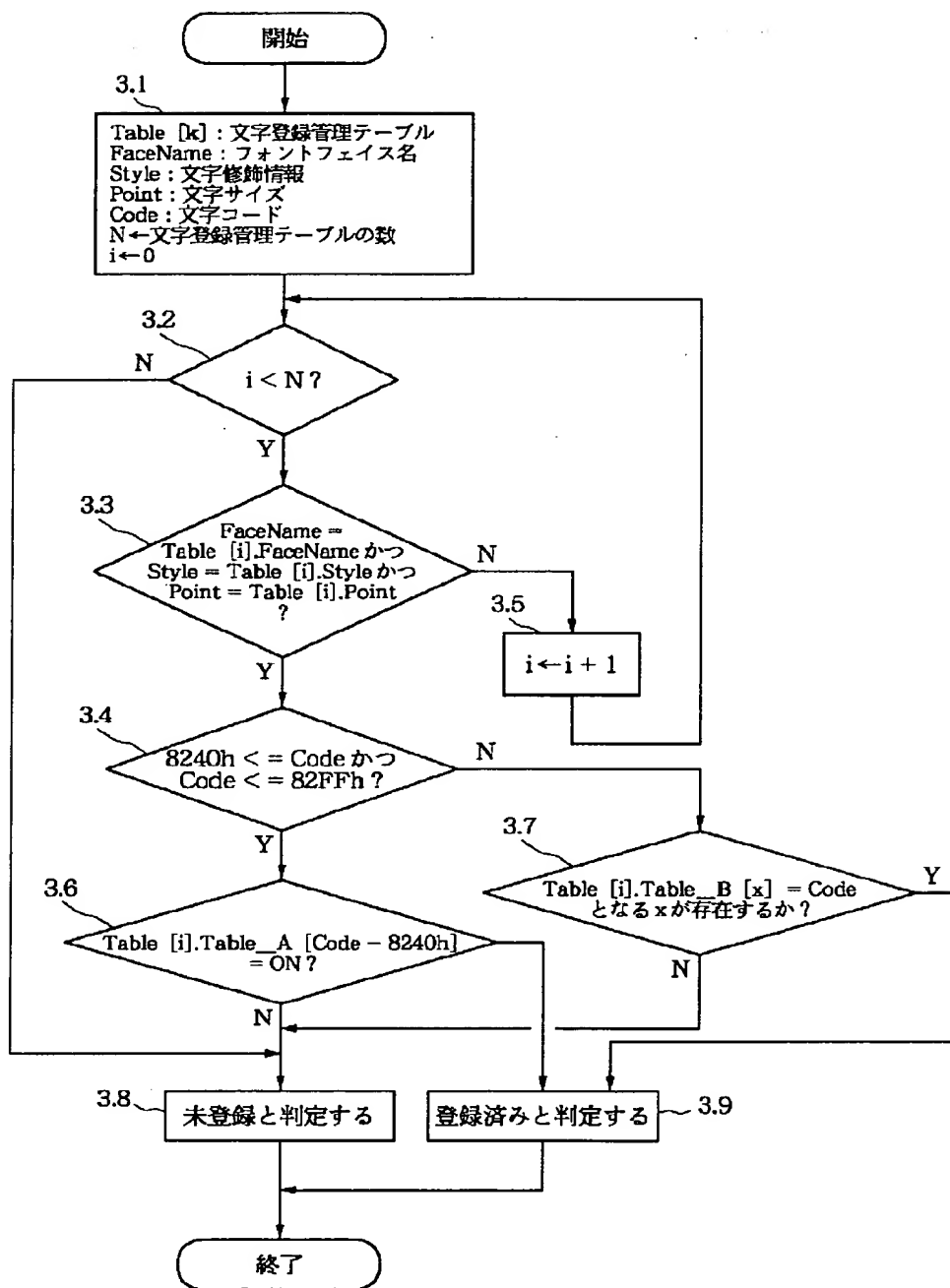
【図 2】



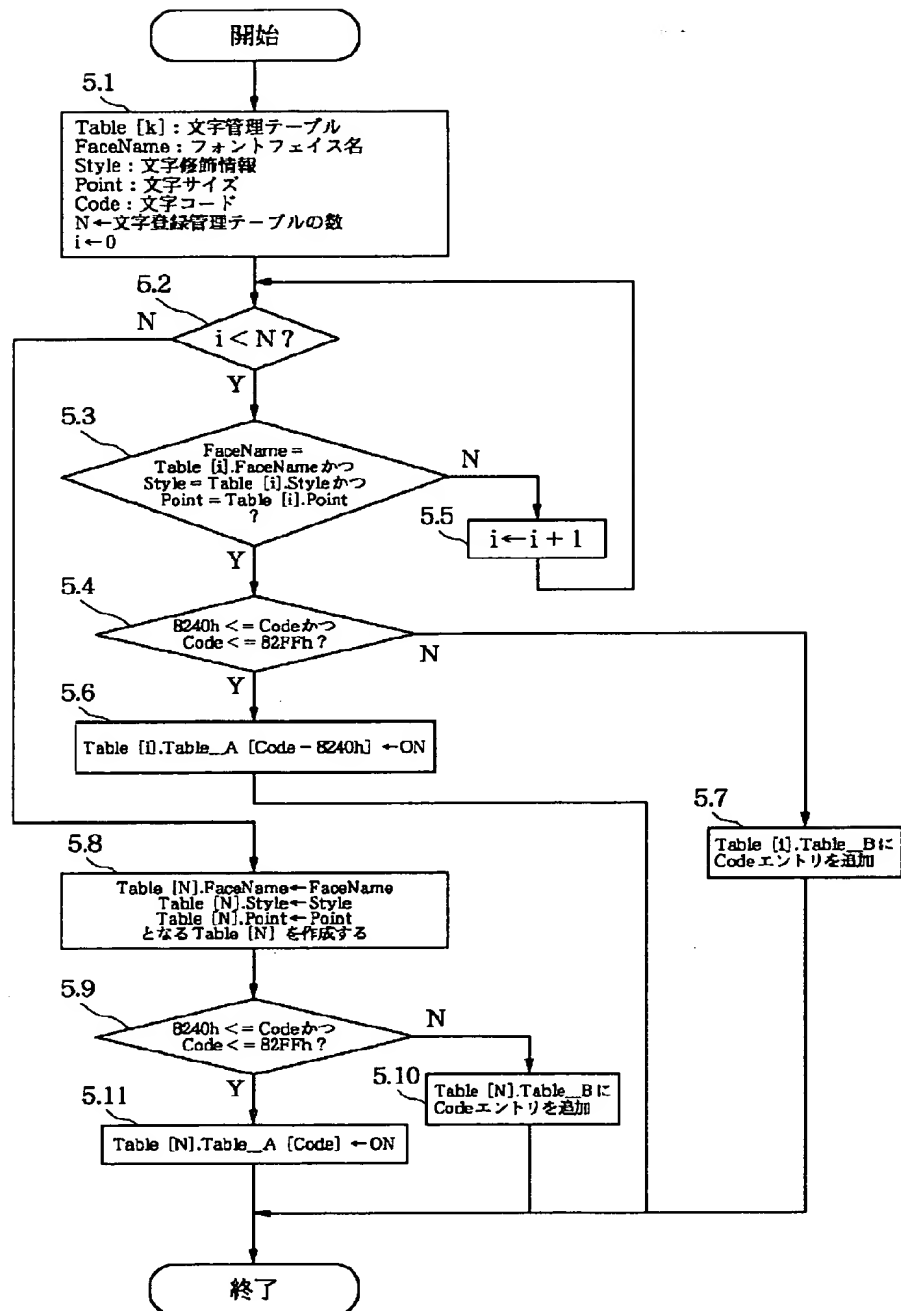
【図 4】

| | |
|---------|------------------------------|
| Table_A | 8240h-82FFhの文字コードに対応した192ビット |
| Table_B | Table_A以外の文字コードを格納するエリア |

【図 3】



【図 5】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.